통보전달대면부(MPI)
사용지도서

국가과학원 콤퓨터과학연구소
우리식조작체계개발조

주체 98(2009)년 4 월

차 례

머리말

개인용콤퓨터의 성능이 비약적으로 높아졌지만 여전히 한대의 개인용콤퓨터로 풀수 없는 방대한 계산량을 가지는 대규모과학기술계산문제들이 수많이 제기되고있다. 이로부터 세계적으로 병렬처리에 대한 연구사업에 많은 노력이 돌려지고있다. 최근에는 망통신속도가 비약적으로 갱신되면서 국부망으로 련결된 개인용콤퓨터들을 가지고 병렬처리를 하는것이 가능하게 되였으며 이것이 하나의 추세로 되고있다.

《통보전달대면부》는 개인용콤퓨터들을 국부망으로 련결한 병렬 및 분산계산환경에서 통보전달(Massage Passing)방식의 병렬프로그람을 개발하고 실행시키기 위하여 제작된 병렬프로그람개발환경이다.

통보전달방식의 병렬프로그람은 C, C++언어로 작성된 순차프로그람코드에 병렬성을 추가하기 위한 통보전달서고함수들을 삽입하는 방법으로 작성한다. 대규모과학기술계산환경 《통보전달대면부》는 통보전달서고함수들과 그에 기초하여 작성된 병렬프로그람들을 콤파일, 련결하여 실행프로그람으로 만들기 위한 콤파일지령들과 실행프로그람을 국부망의 콤퓨터들에서 병렬로 실행시키기 위한 실행지령으로 구성된 병렬프로그람개발 및 실행환경이다.

통보전달대면부를 리용하여 병렬프로그람을 개발하면 한대의 콤퓨터에서는 계산시간이 방대하여 해결하기 힘든 대규모과학기술계산문제를 여러대의 콤퓨터로 협동처리함으로써 해당 문제를 보다 짧은 시간내에 풀수 있다.

류체력학적현상들의 모의 및 해석, 기상예보를 위한 대기층수치해석, 각종 기계설계 및 동작모의, 분자동력학계산, 단백질구조해석 등 많은 과학기술계산분야들에서 병렬계산처리를 진행하는데 효과적으로 리용할수 있다. 대규모과학기술계산을 전문으로 하는 과학자, 기술자, 교원, 학생들이 리용할수 있다.

사용자들은 《통보전달대면부》를 리용하여 대규모과학기술계산문제들을 국부망환경에서 풀수 있는 통보전달(Massage Passing)방식의 병렬프로그람들을 콤파일 및 련결, 실행시킬수 있다.

1. 통보전달대면부의 조작체계요구와 지원환경

통보전달대면부는 《붉은별》사용자용체계 2.0에서 동작한다.

이 프로그람은 병렬계산처리에 참가하는 국부망안의 모든 콤퓨터들이 서로 원격실행을 할수 있도록 매개 콤퓨터에 원격쉘봉사기(rsh-server)와 원격쉘의뢰기(rsh) 프로그람들이 설치된 조건에서 동작할수 있다. 또한 개발하려는 병렬프로그람에 대응하는 순차프로그람의 언어에 따라서 C, C++프로그람개발환경들이 보장되여야 한다.

2. 설치방법

2.1 설치형태

대규모과학기술계산환경 《통보전달대면부》는 2가지 방법으로 설치할수 있다.

기본설치방법은《붉은별》사용자용체계 2.0에서 《통보전달대면부》패키지와 그와 련관되는 다른 프로그람패키지들을 모두 repo형식으로 묶어서 설치한다.

다른 방법은 지령행우에서 rpm 지령으로 개별적인 rpm 패키지들을 따로 따로 설치하는 방법으로 할수도 있다.

2.2 통보전달대면부설치방법

① repo에 의한 설치

《붉은별》사용자용체계 2.0이 설치된 콤퓨터에서 다음과 같은 방법으로 프로그람을 설치한다.

우선 repo설치패키지에서 아래의 그림 1과 같이 mpich.repo 파일을 실행한다.
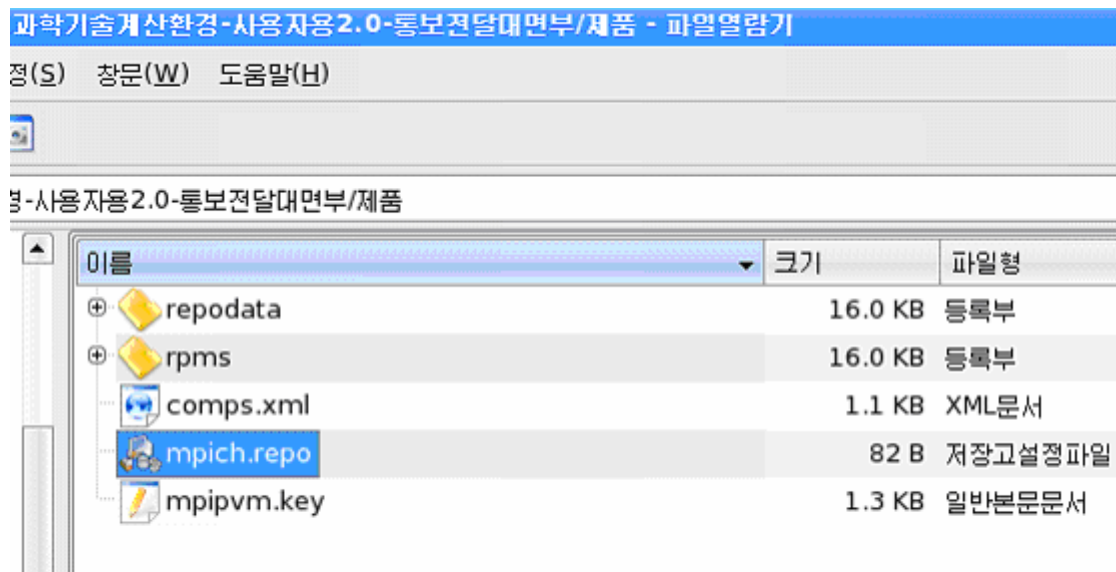
그림 1. repo 에 의한 프로그람 실행

그러면 아래의 그림 2 와 같은 《통보전달대면부-설치》대화창문이 나온다. 이 대화창의 아래에 있는 《다음》단추를 누르면 그림 3 과 같은 《쏘프트웨어 선택》대화칸이 나오는데 《통보전달대면부》항목안에 있는 mpich, rsh, rsh-server 패키지들과 《기타》항목에 있는 gcc, gcc-devel,… 등의 패키지들을 모두 선택한다. 그리고 《다음》단추를 누르면 통보전달대면부와 련관프로그람묶음들인 원격쉘프로그람들이 설치된다 (그림 4).
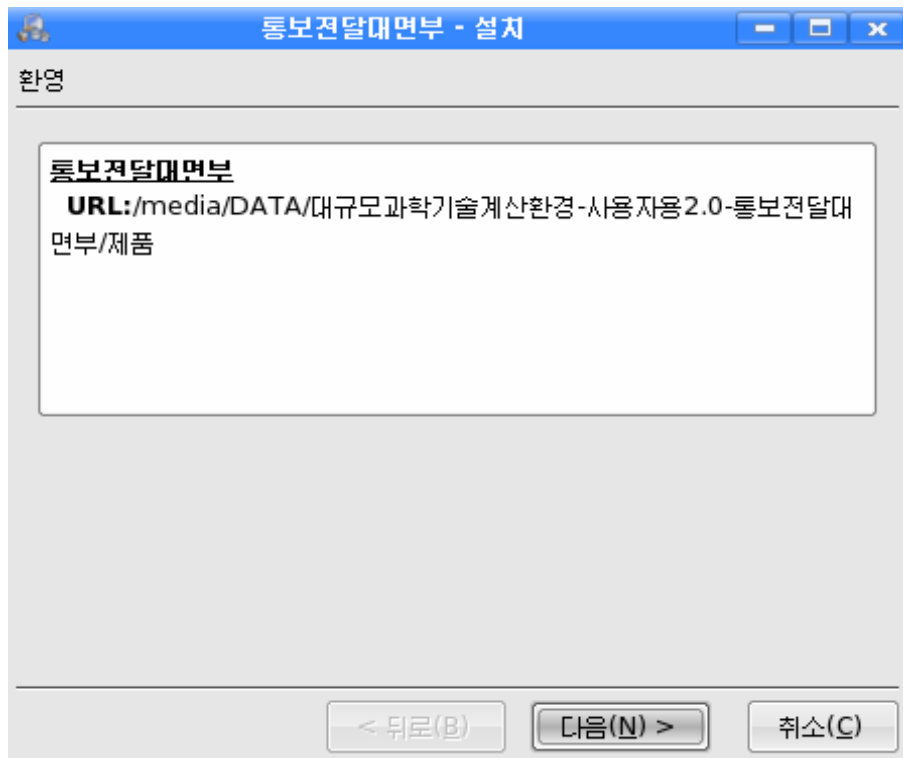
그림 2. 《통보전달대면부》프로그람묶음의 설치시작
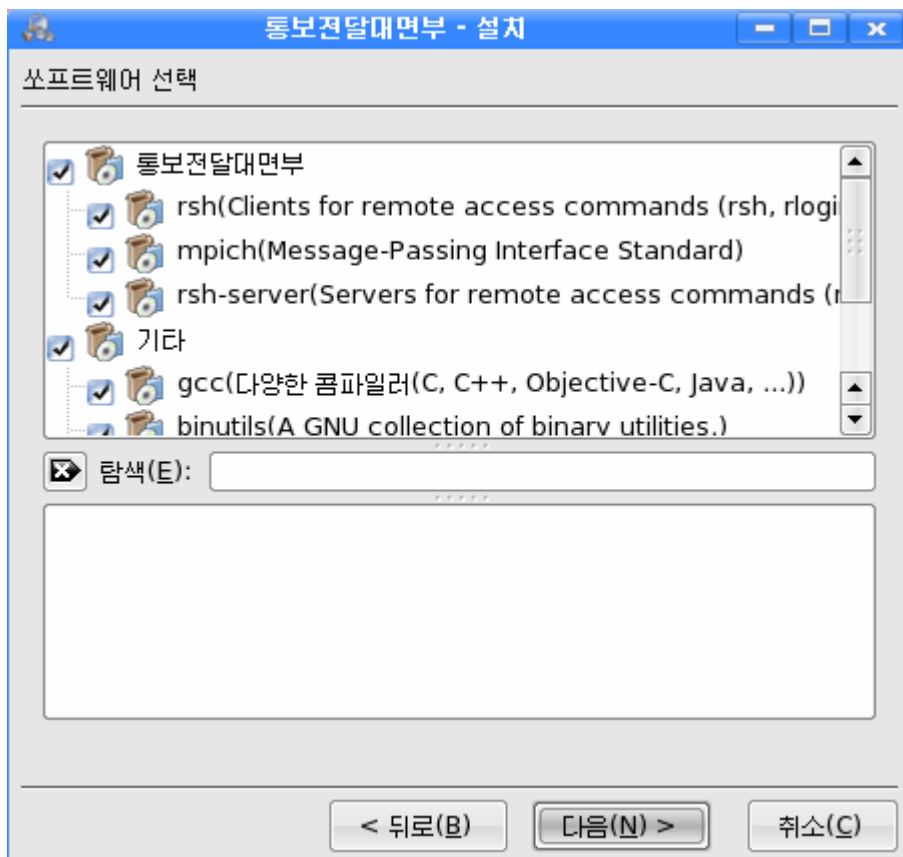
그림 3. 《통보전달대면부》프로그람묶음의 선택대화창
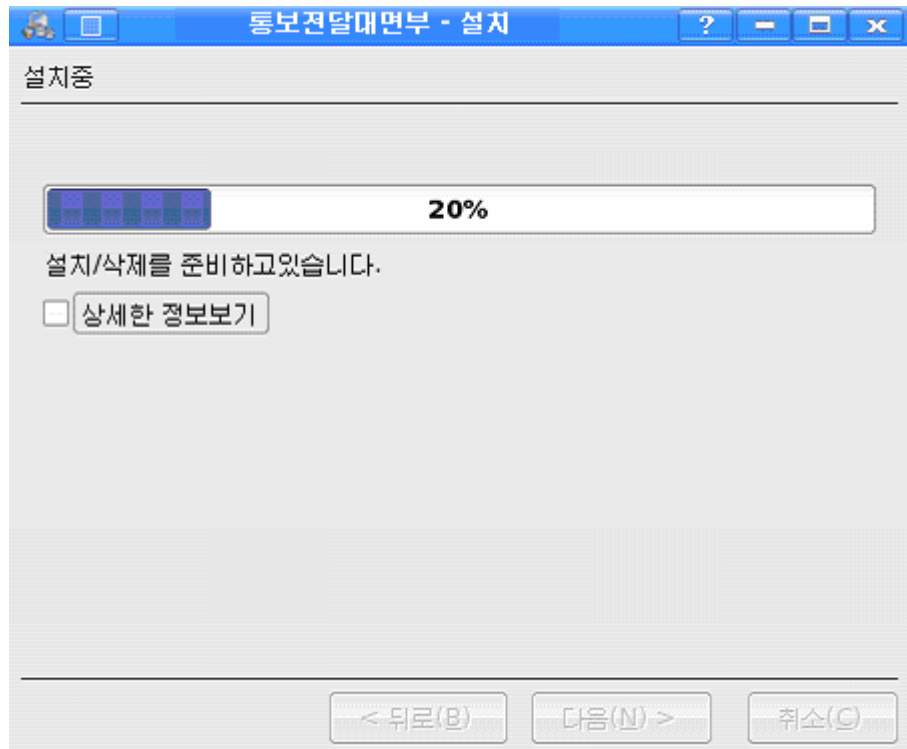
그림 4. 《통보전달대면부》프로그람묶음의 설치과정

지금까지 진행한 《통보전달대면부》프로그람묶음설치과정을 병렬계산에 참가시키려는 국부망안의 모든 콤퓨터들에서 반복한다.

통보전달대면부를 삭제하기 위하여서는 《쏘프트웨어 선택》 대화칸에서 선택하였던것을 다시 해제하고 《다음》단추를 누르면서 나가면 된다.

② rpm 에 의한 설치

《붉은별》사용자용체계 2.0 에 배포된 통보전달대면부패키지는 mpich-1.2.5-1.rs2.0.i386.rpm 이다.

통보전달대면부패키지를 조작탁의 지령행우에서 설치하는 방법은 다음과 같다.

먼저 우의 mpich-1.2.5-1.rs2.0.i386.rpm 패키지를 작업등록부(아래의 실례에서는 /home)에 놓은 다음 rpm 지령으로 설치한다. 즉

```
[root@node0 home]# rpm -Uvh mpich-1.2.5-1.rs2.0.i386.rpm
```

우의 통보전달대면부패키지외에도 다음의 추가적인 원격쉘프로그람(봉사기, 의뢰

기)들이 설치되여야 한다.

먼저 조작탁의 지령행우에서 원격쉘봉사기패키지인 rsh-server-0.17-34.1.386.rpm 을 설치한다. 즉

```
[root@node0 home]# rpm -Uvh rsh-server-0.17-34.1.i386.rpm
```

다음 원격쉘의뢰기패키지인 rsh-0.17-34.1.i386.rpm 을 설치한다. 즉

```
[root@node0 home]# rpm -Uvh rsh-0.17-34.1.i386.rpm
```

또한 gcc 패키지와 그 의존패키지들을 설치하려면 다음과 같이 한다.

```
[root@node0 home]# rpm -Uvh binutils-2.17.50.0.3-6.rs2.0.i386.rpm
[root@node0 home]# rpm -Uvh libgomp-4.1.1-30.rs2.0.i386.rpm
[root@node0 home]# rpm -Uvh glibc-headers-2.5-3.rs2.0.i386.rpm
[root@node0 home]# rpm -Uvh glibc-devel-2.5-3.rs2.0.i386.rpm
[root@node0 home]# rpm -Uvh gcc-4.1.1-30.rs2.0.i386.rpm
```

우에서 진행한 통보전달대면부패키지와 원격쉘패키지, gcc 패키지들의 설치를 병렬계산에 참가시키려는 국부망안의 모든 콤퓨터들에서 반복한다.

실례로 콤퓨터 node1 에서는 다음과 같다.

```
[root@node1 ~]# rpm -Uvh mpich-1.2.5-1.rs2.0.i386.rpm
[root@node1 ~]# rpm -Uvh rsh-server-0.17-34.1.i386.rpm
[root@node1 ~]# rpm -Uvh rsh-0.17-34.1.i386.rpm
[root@node1 ~]# rpm -Uvh binutils-2.17.50.0.3-6.rs2.0.i386.rpm
[root@node1 ~]# rpm -Uvh libgomp-4.1.1-30.rs2.0.i386.rpm
[root@node1 ~]# rpm -Uvh glibc-headers-2.5-3.rs2.0.i386.rpm
[root@node1 ~]# rpm -Uvh glibc-devel-2.5-3.rs2.0.i386.rpm
[root@node1 ~]# rpm -Uvh gcc-4.1.1-30.rs2.0.i386.rpm
```

모든 계산마디점들에서 통보전달대면부가 정확히 설치되면 매개 마디점의 /usr/local 안에 mpich 등록부가 새로 생기고 그 등록부안에 통보전달대면부와 관련한 내용

들이 들어간다. “시작” -> “응용프로그람” -> “대규모과학기술계산환경 《병진》1.0” -> “통보전달대면부” 항목이 생기고 여기에 기본적인 도움말문서들이 련결된다. 또한 탁상면에 MPI 그림기호가 새롭게 생기는것을 볼수 있다. 기타 원격쉘이나 C 프로그람개발환경들도 자기의 기정경로에 설치된다.

모든 콤퓨터들에서 통보전달대면부와 련관프로그람들을 설치한 다음에는 매개 콤퓨터에서 병렬 및 분산계산환경을 구성해주어야 한다.

2.3 병렬 및 분산계산환경의 구성

2.3.1 병렬 및 분산계산을 위한 망장치구성

병렬 및 분산계산을 위한 장치환경은 여러대의 콤퓨터들을 고속이써네트망으로 련결하여 간단히 구축할수 있다. 망에 련결되는 콤퓨터수에는 제한이 없다. 사용자들이 가능하다면 수십, 수백, 수천대까지도 련결할수 있다. 이런 류형을 분산기억다중콤퓨터방식이라고 한다. 여기서 개별콤퓨터들은 서로 기종이 달라도 되며 최소한 《붉은별》사용자용체계 2.0이 설치되고 기동할수 있어야 한다.

《통보전달대면부》는 또한 대칭형다중처리기(SMP) 혹은 Dual-Core와 같은 공유기억다중처리기방식의 병렬처리기들에서도 잘 동작한다. SMP나 Dual-Core와 같은 콤퓨터들은 그 자체가 하나의 병렬처리기로 되며 그것들을 고속망으로 련결하여 보다 큰 규모의 병렬계산환경(분산기억다중콤퓨터방식)을 구성할수도 있다. 여기서 고속망은 100Mbps 또는 1Gbps이상의 전송속도를 가지는 망장치들(스위치하브, 망대면카드)로 련결된 망을 의미한다. 고속망으로 련결되여 하나의 거대한 병렬 및 분산계산환경을 이루는 개별적인 콤퓨터들을 일반적으로 마디점이라고 부른다.

공유기억다중처리기방식은 여러개의 처리기(CPU)들이 한개의 주기억을 공유하고 있는 구성방식이며 분산기억다중콤퓨터방식은 주기억들이 국부망에 분산되여있는 구성방식이다.

2.3.2 원격실행을 위한 망구성

《통보전달대면부》를 리용하여 병렬 및 분산계산을 하자면 원격실행환경이 보장되여야 한다. 이를 위해서 콤퓨터들이 서로 원격실행을 할수 있도록 론리적인 망구성

을 해주어야 한다. 그러자면 모든 콤퓨터에서 /etc/hosts 파일을 편집하여 마디점구성을 해주어야 한다.

일반적으로 체계를 설치할 때 망구성상태는 /etc/hosts 파일에 반영된다.

초기에 /etc/hosts 파일은 다음과 같이 되여있다.

… … …

127.0.0.1   node10.pccluster.org   localdomain.localhost localhost

여기서 node10.pccluster.org 는 《붉은별》사용자용체계를 설치할 때 입력해준 콤퓨터의 이름이다.

우의 내용을 다음과 같이 수정해주어야 한다.

… … …

127.0.0.1   localdomain.localhost localhost

200.200.200.110 node10.pccluster.org node10

여기서 200.200.200.110 은 국부망안에서 이 콤퓨터의 IP 주소이다.

node10.pccluster.org 는 체계설치시 입력해준 콤퓨터이름이고 node10 은 콤퓨터의 가명이다.

다음으로 서로 통신하려는 다른 콤퓨터들의 IP 주소와 콤퓨터이름을 서술하는 행들을 다음행부터 련이어 추가하여 준다. 즉 아래와 같다.

127.0.0.1 localdomain.localhost localhost

200.200.200.110 node10.pccluster.org node10

200.200.200.111 node11.pccluster.org node11

200.200.200.112 node12.pccluster.org node12

200.200.200.113 node13.pccluster.org node13

⋮

이렇게 해주어야 node10, node11, node12, node13, … 등의 이름을 가진 콤퓨터들에 대해 원격실행을 위한 주소해결이 가능하게 된다. /etc/hosts 파일에 기록되지 않은 마디점에 대해서는 장치적으로 망에 련결되여있다고 하여도 원격실행을 할수 없다.

우의 /etc/hosts 파일편집과정을 병렬처리에 참가하는 모든 콤퓨터들에서 반복한다. 파일의 내용은 모든 콤퓨터에서 다 꼭같이 해준다. 이와 같은 방법으로 통보전달대면부를 리용하여 병렬계산을 하려는 모든 콤퓨터들에 원격실행이 가능하도록 망구성을 해준다.

2.3.3 보안설정해제와 원격실행환경의 활성화

병렬계산을 하자면 계산에 참가하는 모든 콤퓨터들에서 방화벽이 해제되여야 한다. 체계의 설치시 혹은 설치후에 모든 마디점들에서 보안기능을 "비설정"으로 해준다.

다음 아래의 과정을 차례로 수행하여야 한다.

- 원격실행환경을 보장할수 있는 구성파일을 수정 또는 생성해주어야 한다.

① /etc/hosts.equiv 파일을 새로 생성한다.

본문편집기에서 새로운 파일을 하나 열고 파일이름을 hosts.equiv 로 달아준 다음 /etc 안에 보관한다. 이 파일에는 한행에 한개 콤퓨터씩 입력하여 서로 통신하게 하려는 콤퓨터들의 이름을 전부 기록한다(가명만 주어도 된다.).

실례로 /etc/hosts.equiv 파일의 내용을 다음과 같이 구성할수 있다.

```
node10
node11
node12
node13
   ⋮
```

여기서 node10, node11, node12, node13 … 등은 콤퓨터이름(가명)이다.

우의 /etc/hosts.equiv 파일편집과정을 병렬처리에 참가하는 모든 콤퓨터들에서 반복한다. 파일의 내용은 모든 콤퓨터에서 다 꼭같이 해준다.

② $HOME/.rhosts 파일을 새로 생성한다.

여기서 $HOME 은 root 환경(관리자환경)이면 /root 이고 리용자환경(실례로 user1)이면 /home/user1 과 같이 된다.

$HOME/.rhosts 파일의 내용은 /etc/hosts.equiv 파일과 꼭같다.

우의 $HOME/.rhosts 파일편집 및 생성과정 역시 hosts, hosts.equiv 파일들과 마찬가지로 병렬처리에 참가하는 모든 콤퓨터(마디점)들에서 반복한다.

- 통신과 원격실행환경에 영향을 주는 모든 보안기능들을 해제해야 한다.

① /etc/xinetd.d/rlogin 파일과 /etc/xinetd.d/rsh 파일을 임의의 편집기로 열기한 다음 그 내용을 다음과 같이 수정하고 보관한다.

이 파일에서 disable = yes 로 되여있는것을 disable = no 로 바꾸어준다.

/etc/xinetd.d/안의 rlogin, rsh 파일내용수정을 병렬처리에 참가하는 모든 콤퓨터(마디점)들에서 반복한다.

② /etc/pam.d/rlogin 파일과 /etc/pam.d/rsh 파일의 내용을 수정한다.

먼저 /etc/pam.d/rsh 파일안에서 다음과 같은 행을 찾는다.

```
…
auth required pam_rhosts_auth.so
…
```

이 행에서 required 를 sufficient 로 변경한 다음 제일 첫행으로 옮긴다. 즉

```
auth sufficient pam_rhosts_auth.so
…
```

다음 /etc/pam.d/rlogin 파일안에서 우와 꼭같은 행을 찾아서 첫행으로 옮긴다. 즉

```
auth sufficient pam_rhosts_auth.so
…
```

우의 /etc/pam.d/rlogin , /etc/pam.d/rsh 파일들의 수정을 병렬처리에 참가하는 모든 콤퓨터(마디점)들에서 반복한다.

③ 모든 콤퓨터(마디점)들에서 xinetd 데몬을 재기동한다.

실례로 콤퓨터 node10 우에서의 재기동지령은 다음과 같다.

[root@node10 ~]# /etc/rc.d/init.d/xinetd restart

혹은

[root@node10 ~]# service xinetd restart

이때 xinetd 데몬이 우에서 변경시킨 새로운 설정내용들을 가지고 재기동하게 된다. 이미 기동되여있는 xinetd 데몬이 정지되고 다시 xinetd 데몬이 기동하면서 다음의 통지문을 내보낸다.

xinetd 중지합니다: [확인]

xinetd 봉사를 시작합니다: [확인]

2.3.4 원격실행환경의 검사

원격실행환경을 구축한 다음에는 그것이 정확한가를 반드시 확인하여야 한다.

조작탁의 지령행에서 병렬계산에 참가하는 모든 콤퓨터들에 원격쉘지령을 실행해보는 방법으로 검사한다.

실례로 콤퓨터 node10 에서 node11 에 대한 원격실행상태를 확인하자면 조작탁의 지령행에서 아래와 같이 한다.

[root@node10 ~]# rlogin node11

또는

[root@node10 ~]# rsh node11

이때 아무런 암호도 물어보지 않고 원격콤퓨터(node11)에로의 마지막 가입자(마디점이름), 가입날자와 시간정보가 나오고 그 마디점에서의 지령입력재촉상태에 들어가면 이것은 원격실행환경이 보장된것이다.

원격실행상태는 임의의 마디점에서 rsh node11 ls 와 같은 지령을 입력하여 콤퓨터 node11 의 파일내용을 보는 방법으로 검사할수도 있다.

2.3.5 기계정보파일의 구성

이 단계는 통보전달대면부병렬프로그람들이 실행되야 할 기계들의 목록을 제시해주는 과정이다.

모든 마디점들에 통보전달대면부를 설치하면 /usr/local/안에 mpich라는 등록부가 새로 생기고 그안의 부분등록부들에 여러가지 실행프로그람들과 통신서고를 비롯하여 통보전달대면부와 관련되는 내용들이 들어간다.

기계들의 목록은 /usr/local/mpich/share/machines.LINUX파일안에 서술하는 방법으로 작성한다. /etc/hosts.equiv파일과 내용이 비슷하다. 한행에 한개 기계이름을 서술하여야 하며 SMP와 같은 다중처리기들은 기계이름을 준 다음에 두점(:)을 치고 처리기(CPU)개수를 밝혀주어야 한다.

실례로 machines.LINUX 파일의 내용은 다음과 같이 될수 있다.

```
node10:4
node11:4
node12
node13
```

이 실례는 병렬계산에 4개의 콤퓨터(마디점)가 참가하며 그중에서 node10과 node11는 처리기 4개를 장비한 다중처리기마디점이라는것을 가리킨다.

/usr/local/mpich/share/machines.LINUX파일은 반드시 편집해주어야 하며 정확하여야 한다. 앞에서 본 /etc/hosts, /etc/hosts.equiv, $HOME/.rhosts 파일들에 반영된 마디점이라고 하여도 machines.LINUX파일안에 없으면 그 마디점은 병렬계산에 참가하지 못한다. 즉 어떤 마디점이 병렬계산에 참가하자면 /etc/hosts, /etc/hosts.equiv, $HOME/.rhosts, machines.LINUX파일들에 다 반영되여있어야 한다. 다시말하여 /etc/hosts, /etc/hosts.equiv, $HOME/.rhosts파일들에 반영된 마디점들은 순수원격쉘만을 위한것이고 machines.LINUX파일에 반영된 마디점들은 통보전달대면부체계 자체가 별도로 병렬프로쎄스들의 생성과 실행을 위하여 참고하는 기계들의 목록이다.

원격실행환경구성과는 달리 machines.LINUX파일은 모든 마디점들에 다 없어도 된다. 사용자가 병렬프로그람을 처음 기동시키는 주마디점에만 하나 있으면 된다.

2.3.6 통보전달대면부프로그람실행을 위한 쉘실행환경변수의 설정

《통보전달대면부》는 병렬프로그람을 콤파일하고 실행시키기 위한 여러가지 쉘 각본지령들을 제공하였다. 이것들은 《통보전달대면부》를 설치하면 /usr/local/mpich /bin/안에 놓이는데 이 설치경로가 반드시 쉘의 실행환경변수인 PATH에 반영되여야 쉘이 이 지령들을 찾아서 실행할수 있다. 그렇치 않으면 해당 쉘은 통보전달대면부와 관련된 지령들을 찾지 못하며 결국 오유통보를 내보낸다.

《붉은별》체계에서 기본 많이 리용되고있는 쉘은 bash이다.

통보전달대면부지령들을 실행하기 위한 쉘의 실행환경변수 PATH의 설정은 /etc/ bashrc파일안에 PATH설정지령행을 하나 삽입하는 방법으로 한다. 즉 bashrc파일을 본문편집기로 열고 실행환경변수설정지령을 삽입해준다.

구체적으로 보면 초기에 etc/bashrc파일의 내용은 아래와 같다.

```
# /etc/bashrc
# System wide functions and aliases
…
fi
# Vim : ts=4 : sw =4
```

여기에 다음의 행을 처음 혹은 마지막에 삽입해주고 보관한다.

```
export PATH=/usr/local/mpich/bin:$PATH
```

이렇게 해주면 체계에서 기동되는 모든 bash들이 통보전달대면부지령들의 경로를 정확히 찾아서 실행할수 있게 된다.

우의 export지령을 조작탁의 현재 bash쉘의 지령행우에서 직접 실행하는 방법으로 할수도 있다. 이 경우에는 바로 그 bash쉘대화기에서만 통보전달대면부관련

지령들이 실행되고 다른 대화기들에서는 실행되지 않는다.

쉘실행환경변수 PATH의 설정은 통보전달대면부와 관련된 모든 지령들을 실행하기 위한 필수적인 설정항목이다. 쉘실행환경변수설정은 국부망안의 모든 콤퓨터들에서 하지 않아도 된다. 병렬프로그람을 콤파일, 련결, 실행시키는 주콤퓨터에서만 설정하면 된다.

3. 통보전달대면부(MPI)리용방법

《통보전달대면부》를 리용하여 대규모과학기술계산문제들을 병렬처리하자면 기본적으로 3가지 문제가 나선다. 즉 MPI병렬프로그람의 작성, 그것을 실행프로그람으로 만들기 위한 콤파일과 련결, 그리고 MPI병렬프로그람실행방법이다.

《통보전달대면부》는 병렬프로그람개발자들이 병렬프로그람작성시 리용할수 있는 통보전달서고와 이 통보전달서고함수들을 리용하여 작성한 병렬프로그람을 콤파일하기 위한 콤파일러, 그리고 실행가능한 병렬프로그람을 국부망에 속한 여러대의 콤퓨터우에서 병렬로 실행시키기 위한 실행기를 제공한다.

《통보전달대면부》병렬프로그람작성을 위한 전용의 편집기는 따로 없다. 《붉은별》조작체계에서 제공하는 임의의 편집기들을 다 리용할수 있다. C, C++순차프로그람에 통보전달서고함수들을 삽입하여 병렬프로그람을 작성하는 방법은 *MPI 프로그람작성지도서*에서 상세히 설명한다.

이 문서(*MPI 사용지도서*)에서는 이미 작성되여있는 통보전달대면부병렬프로그람을 콤파일하고 련결하여 실행파일로 만드는 방법, 실행가능한 병렬프로그람을 실행시키는 방법에 대하여 설명한다.

3.1 통보전달대면부프로그람의 콤파일 및 련결

통보전달대면부병렬프로그람의 콤파일과 련결은 조작탁의 지령행우에서 간단히 할수 있다.

통보전달대면부는 C, C++형식의 MPI병렬프로그람들을 콤파일하고 련결할수 있는 쉘각본지령들을 제공하였다. 즉 순차프로그람이 C프로그람인 경우에 대해서는 mpicc, C++에 대해서는 mpiCC 이다.

실례로 《통보전달대면부》를 설치할 때 /usr/local/mpich/examples등록부안에

기정으로 들어가는 병렬프로그람인 cpi.c를 콤파일하는 방법은 아래와 같다.

```
[root@node0 ~]# cd /usr/local/mpich/examples
[root@node0 examples]# mpicc -c cpi.c
```

우의 콤파일지령이 실행되면 그 결과로 cpi.c가 있는 위치에 객체파일 cpi.o가 생성된다.

콤파일결과로 얻어진 중간코드로부터 실행파일을 만드는 지령은 다음과 같다.

```
[root@node0 examples]# mpicc -o cpi cpi.o
```

이 과정에 통보전달서고를 비롯한 각종 서고함수들이 객체파일에 련결되여 결과적으로 실행가능한 파일 cpi가 생성된다.

아래와 같이 콤파일과 련결을 한개의 지령에서 할수도 있다.

```
[root@node0 examples]# mpicc -o cpi cpi.c
```

이렇게 하면 원천프로그람 cpi.c로부터 실행가능한 병렬프로그람 cpi가 얻어지게 된다.

3.2 통보전달대면부프로그람실행

실행가능한 통보전달대면부병렬프로그람을 실행하기 위하여 mpirun지령을 사용한다. 이 지령은 《통보전달대면부》를 설치하면 /usr/local/mpich/bin/안에 들어간다.

이 지령을 사용하여 우에서 얻어진 실행가능한 병렬프로그람 cpi를 6개의 처리기우에서 실행시키는 방법은 아래와 같다.

```
[root@node10 ~] # cd /usr/local/mpich/examples
[root@node10 examples] # mpirun -np 6 cpi
```

여기서 알수 있는것처럼 -np추가선택에 6이라는 값을 주어서 병렬프로그람 cpi를 6개의 프로쎄스로 실행할것을 지적하였다. 파라메터 cpi는 mpirun지령에 의해 실행되여야 할 실행가능한 병렬프로그람의 이름이다. 이렇게 지령을 주었을 때 기계구성이 우의 *2.3.5 기계파일의 구성* 절에서 실례를 든것처럼 되여있다면 cpi프로그람이 node10에서 4개, node11에서 2개의 프로쎄스로 생성되여 총 6개의 프로쎄스가 병렬로 실행되게 된다(node10, node11은 SMP처리기).

만일 node10, node11콤퓨터들이 SMP처리기가 아니라면 기계정보파일에 들어있는 4개의 콤퓨터에서 다 병렬프로쎄스가 생성되는데 node10에서 1개, node11에서 2개, node12에서 2개 , node13에서 1개가 각각 생성되여 총 6개의 병렬프로쎄스가 생성, 실행되게 된다.

mpirun -np 6 cpi 지령은 주마디점에서 한번만 해주면 되는데 주마디점은 기계정보파일에 포함되여있는것중에서 임의로 선택할수 있다. 우에서는 node10이 주마디점으로 된다. 이렇게 병렬프로그람을 처음 기동시키는 마디점을 주콤퓨터 혹은 주마디점이라고 부른다. 주마디점에는 반드시 기계정보파일이 있어야 한다(절 2.3.5).

MPI병렬프로그람을 실행하는데서 한가지 중요한것은 실행가능한 병렬프로그람이 병렬계산에 참가하는 매개 마디점들에서 꼭같은 경로안에 있어야 한다는것이다. 우의 실례에서 cpi는 매개 마디점에서 꼭같이 /usr/local/mpich/examples경로에 있어야 한다. 이것은 rcp지령을 리용하여 실행프로그람을 원격마디점들에 복사하는 방법으로 진행한다. 실례로 실행프로그람 cpi가 node10에서 콤파일, 련결하여 얻어졌을 때 그것을 node11의 동일한 위치에 복사하는 방법은 다음과 같다.

```
[root@node10 ~] # rcp /usr/local/mpich/examples/cpi node11: /usr/local/mpich/examples
```

혹은 현재의 작업등록부가 /usr/local/mpich/examples이면 아래와 같이 할수 있다.

```
[root@node10 examples]  # rcp cpi node11:/usr/local/mpich/examples
```

cpi를 복사해야 할 마디점이 여러개인 경우에는 우의 지령행에서 node11을 node12, node13, … 등으로 바꾸어주면서 복사하면 된다.

우의 mpirun지령형식은 통보전달대면부병렬프로그람들을 실행시키기 위한 기본형식이다. mpirun지령의 사용방법만을 알고싶다면 아래와 같이 지령을 입력하면 된다.

[root@node10 ~] mpirun -help

…

mpirun지령의 사용방법에 대한 설명

…

mpirun지령에는 우에서 본 -np 나 -help외에도 여러가지 추가선택들이 함께 지정될수도 있는데 mpirun -help 지령을 실행할 때 나오는 설명을 통하여 상세하게 알수 있다.

mpirun지령에 넘기는 파라메터들이 잘못되였거나 없는 프로그람이름이 지정되는 경우에는 해당한 오유통보를 내보내고 프로그람실행을 취소한다.

3.3 통보전달대면부의 도움말보기

병렬프로그람개발환경《통보전달대면부》에서는 사용자들이 병렬프로그람을 작성하고 그것을 콤파일하고 련결하여 실행파일을 얻고 련이어 그것을 병렬로 실행시킬 때 참고할수 있는 여러가지 풍부한 도움말을 제공하고있다.

/usr/local/mpich/doc/안에 *MPI프로그람작성지도서*, *MPI사용지도서*, *MPI함수설명서*를 주었으며 쉘각본지령들과 통보전달서고안에 있는 수많은 MPI함수들에 대한 도움말이 HTML형식과 man페지형식 두가지로 제공되였다.

HTML문서들은 /usr/local/mpich/www/안에 들어있으며 《내나라열람기》 또는 파일열람기로 열람할수 있다.

man페지형식의 설명서들은 조작탁의 지령행우에서 man지령을 통하여 열람할수 있다. 실례로

[root@node10 examples] # man mpirun

…

mpirun지령의 사용법에 대한 설명
…
[root@node10 examples] # man mpicc
…
mpicc지령의 사용법에 대한 설명
…
[root@node10 examples] # man MPI_Init
…
MPI함수인 MPI_Init()의 기능과 사용법에 대한 설명
…

우와 같은 방법으로 모든 통보전달서고함수들의 기능과 사용법을 알아볼수 있다.

4. 오유통보에 대한 설명

통보전달대면부병렬프로그람을 작성한 다음 그것을 콤파일하고 련결하여 실행파일로 만들며 련이어 실행가능한 병렬프로그람을 실행시키는 전과정에 병렬프로그람개발자는 여러가지 원인으로 많은 오유통보를 받을수 있다.

병렬프로그람개발자들이 많이 범할수 있는 오유형태와 그에 따르는 오유통보와 대책들은 다음과 같다.

- 병렬프로그람을 작성할 때 문법적오유를 범하는 경우

병렬프로그람을 다 편집하였을 때 원천코드에 문법적오유가 있으면 콤파일도중에 정지되며 문법오유가 들어있는 행번호와 오유형태를 통보하고 끝난다. 오유통보문은 해당 프로그람작성언어의 콤파일러(gcc 혹은 g++)가 내보낸다.

이때 병렬프로그람개발자는 원천코드에서 문법오유를 수정한 다음 다시 콤파일하여야 한다.

- 병렬프로그람을 작성할 때 론리적오유를 범하는 경우

병렬프로그람안에 들어있는 론리적오유는 콤파일시에는 발견되지 않으며 실행과정에 발견되게 된다. 실례로 프로그람에서 창조하지 않은 기억구역에 대한 읽기나

쓰기, 존재하지 않는 자료에 대한 송수신요구, 지정된 배렬의 첨수한계를 벗어난 구역에 대한 참조 혹은 송수신요구 등 많은 오유형태가 있으며 이 경우에 병렬프로그램은 실행 도중에 정지되여 해당한 오유통보를 내보낸다. 이때 병렬프로그람 개발자는 원천코드에서 론리적오유를 수정한 다음 다시 콤파일, 련결, 실행하여야 한다.

- 병렬프로그람을 콤파일, 련결할 때 지령추가선택이나 파라메터값을 틀리게 입력하는 경우

병렬프로그람콤파일러지령에 지정해준 이름의 원천프로그람이 없는 경우에 《그와 같은 파일이나 등록부는 없습니다.》라는 통보가 나온다. 이때 사용자는 다시 정확한 원천프로그람이름을 지정하여 콤파일, 련결을 진행하여야 한다.

콤파일과 련결을 한번에 할 때 생성되여야 할 실행파일에 붙여질 이름을 지정하지 않은 경우 오유통보가 나온다. 이 경우에는 원천파일이 현재의 작업등록부에서 없어지는 결과를 초래한다. 사용자는 다시 원천코드를 작성하여 실행프로그람의 이름을 반드시 지정하고 콤파일, 련결을 하여야 한다.

- 병렬프로그람을 실행할 때 지령추가선택이나 파라메터값을 틀리게 입력하는 경우

병렬프로그람실행지령에 아무런 추가선택이나 파라메터도 주지 않았거나 또는 실행프로그람의 이름을 지정하지 않은 경우에 《프로그람이름을 지정하지 않았습니다.》라는 오유통보를 내보낸다. 이때 사용자는 실행프로그람이름을 정확히 지정해주면서 다시 실행하여야 한다.

병렬프로그람실행지령에 프로쎄스개수값을 부의 옹근수로 지정하는 경우(실례로 -np -4로 한 경우)에《 xx는 의미가 없는 처리기수입니다. 끝냅니다.》라는 오유통보를 내보낸다. 이때 사용자는 프로쎄스개수값을 정확한 정의 옹근수값으로 지정하여 실행을 다시 하여야 한다.

- 원격실행환경을 정확히 구성하지 못한 경우

어떤 마디점이 원격쉘구성에서 배제되여있으나 기계정보파일에는 반영되여있을 때 그 마디점을 병렬계산에 참가시키는 경우 오유통보가 나온다. 먼저 프로쎄스그룹이 현시되고 그 마디점에 대해 이름에 의한 호스트얻기를 할수 없으며 무효한 이름이라는 오유통보가 나온다. 이때는 /etc/hosts, /etc/equiv, /root/.rhosts 파일들을 다시

열고 해당마디점을 웬격쉘구성에 포함시켜야 한다(절 2.3.2 , 2.3.3 참고).

원격쉘재구성을 모든 마디점에서 반복해주어야 한다.

- C , C++ 프로그람개발환경이 보장되지 않은 경우

C, C++프로그람개발환경이 보장되지 않으면 병렬프로그람콤파일단계에서 mpicc 혹은 mpiCC 를 실행할 때 <cc , 혹은 gcc 지령을 찾을수 없다.>는 내용의 오유통보가 나온다. 이때는 통보전달대면부관련패키지설치시 “기타”항목안의 패키지들을 다시 확인하고 설치한 다음 다시 시도해야 한다.

- 통보전달대면부지령들을 쉘이 찾지 못하는 경우

통보전달대면부병렬프로그람을 콤파일하고 실행하기 위하여 **mpicc** 혹은 **mpirun** 지령을 실행하였을 때

bash : mpicc : command not found

혹은

bash : mpirun : command not found

와 같은 통보들이 나올수 있다.

이것은 bash 가 통보전달대면부지령들을 찾지 못하여 내보내는 오유통보이다.

이때는 bash 의 환경변수 PATH 에 통보전달대면부지령들의 위치를 설정해주어야 한다. 절 2.3.6 을 참고하여 PATH 를 설정한 다음에 다시 지령들을 실행하여야 한다.

- 기계정보파일의 내용이 비여있는 경우

사용자가 프로그람을 실행시키기 위하여 mpirun 지령을 실행하면 병렬프로그람 실행기는 병렬프로쎄스들을 실행할 콤퓨터들의 이름을 기계정보파일에서 얻는다. 그런데 여기에 콤퓨터이름들이 전혀 기록되여있지 않을 때는 《xx 개의 프로쎄스들을 실행하기에는 기계가 충분하지 못합니다. 기계정보파일 machines.LINUX 에 마디점을 더 추가해주십시오.》라는 오유통보가 나온다. 이때 사용자는 /usr/local/mpich/share/안에 있는 machines.LINUX 파일을 본문편집기로 열고 기계들의 이름을 더 추가한 다음 보관한다(절 2.3.5). 그다음 다시 병렬프로그람실행지령을 실행하여야 한다.

- 기계정보파일안에 반영된 일부 콤퓨터들이 물리적으로 망련결이 차단되였거나 기동상태에 있지 않을 때 병렬프로그람실행지령을 실행하는 경우

원격접속을 할수 없는 마디점이름이 나오고 경로가 없다는 오유통보가 나온다. 이때는 ping 지령을 리용하여 망의 물리적련결상태를 다시 확인하고 고장을 회복한

다음 다시 실행해야 한다.

- 실행프로그람이 다른 원격마디점의 동일한 경로에 존재하지 않을 때

실행프로그람이 다른 원격마디점의 동일한 경로에 존재하지 않을 때 그 마디점을 병렬계산에 참가시켜 프로그람실행을 시도하면 경로이름과 함께 지정된 실행프로그람이 없다는 오유통보를 내보내고 실행을 중지한다. 잠시후에 정상인 다른 마디점들에서 기동된 프로쎄스들로부터 p4 오유가 나온다. 이때는 실행하려는 병렬프로그람을 계산에 참가하는 모든 마디점들의 꼭같은 경로에 복사해준 다음 다시 실행하여야 한다(절 3.2 참고).

5. 리용상 주의사항

병렬프로그람원천을 콤파일과 련결을 한번에 할 때 실행파일의 이름을 지정하지 않는것은 위험하다. 이 경우 원천파일이 없어지는 결과를 초래한다.

실례로 병렬프로그람원천 /opt/cpi.c 를 콤파일, 련결할 때

[root@node0 opt]# mpicc -o cpi.c

와 같이 지령을 입력하면 콤파일, 련결을 할수 없으며 원천파일 cpi.c 가 없어진다. 이런 경우에 프로그람개발자는 이미 작성하였던 원천코드를 처음부터 다시 작성하여야 한다. 따라서 아래와 같이 새로 생성되는 실행파일의 이름을 반드시 지정해주어야 한다.

[root@node0 opt]# mpicc -o cpi cpi.c